



#22  
wm

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re: Bachmann et al.

Serial No.: 09/303,356

Filed: 04/29/1999

For: Method for deferred  
deletion of entries for a  
directory server backing  
store

§ Group Art Unit: 2172

§

§ Examiner: Nguyen, Tam

§

§ Atty. Docket No.: AT9-98-955

§

Certificate of Mailing  
Under 37 C.F.R. § 1.8(a)

I hereby certify that this correspondence  
is being deposited with the United States  
Postal Service as First Class mail in an  
envelope addressed to:

Mail Stop Appeal Brief--Patent  
Commissioner for Patents  
PO Box 1450

Alexandria, VA 22313-1450  
on December 4, 2003.

By: 

Joseph R. Burwell, Reg. No 44,468

5

APPELLANT'S BRIEF

IN RESPONSE TO OFFICE ACTION UNDER 37 C.F.R. § 1.192

This brief is filed in triplicate in support of the Notice  
of Appeal, filed 09/04/2003, and which appealed from the  
decision of the examiner dated 06/04/2003 rejecting claims  
1-20. The fee required under 37 C.F.R. § 1.17(c) for filing a  
brief in support of an appeal is provided in the Transmittal of  
Appeal Brief filed herewith.

15

RECEIVED

DEC 11 2003

Technology Center 2100

12/10/2003 DTESSEH1 00000107 501888 09303356

02 FC:1402 330.00 DA

1. REAL PARTY IN INTEREST

The real party in interest in this appeal is International Business Machines Corporation (IBM).

5

2. RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

10

3. STATUS OF CLAIMS

Claims 1-20 are pending in this application; claims 1-20 have been finally rejected; claims 1-20 have been appealed. No claims have been canceled, withdrawn, or allowed.

15

4. STATUS OF AMENDMENTS

20

No after-final amendments have been filed.

5. SUMMARY OF INVENTION

The present invention is a method for deferred deletion of entries in a directory service backing store. Although shown as a preferred embodiment within the specification, the invention is not limited to a Lightweight Directory Access Protocol (LDAP) directory service provided with a DB/2 backing store. As stated in the specification, the principles of the

25

present invention may be practiced in other types of directory services, e.g., X.500, and using other relational database management systems, e.g., Oracle, Sybase, Informix, etc., as the backing store.

5        In either the present invention or the prior art, an entry in an LDAP directory is deleted using an SQL statement. In the prior art, the directory server responds to the delete entry statement by instituting a global lock on the database tables to ensure that data in those tables cannot be modified  
10 while the entry is being deleted from the directory. In contrast, the present invention provides an enhanced delete operation whereby the entry is marked for deletion, and the actual deletion is completed at a later time.

More specifically, the present invention is a method for  
15 deleting an entry from a directory in which directory information is stored in a set of database tables; the deletion is initiated in response to a request to delete a directory entry. In response, the directory entry is tagged in some manner as being a deleted entry, preferably by setting  
20 the entry's creation time to a null value. If a search query is received thereafter, the method excludes tagged entries from search results that would otherwise satisfy the search query. At a periodic interval, the routine then searches for tagged entries, and references to the tagged entries are then  
25 deleted throughout the set of database tables. In this manner, the completion of the entry deletion operation is deferred to enable directory queries to be processed even if deleted entries have not yet been fully expunged from the directory.

6. ISSUES

The issues on appeal are:

whether claims 1-13 and 16-18 are unpatentable over Kennedy, "System and method for managing electronic mail messages using a client-based database", U.S. Patent No. 6,134,582, filed 05/26/1998, issued 10/17/2000 (hereinafter Kennedy), in view of Bachmann et al., "Method of hierarchical LDAP searching with relational tables", U.S. Patent No. 6,085,188, filed 03/30/1998, issued 07/04/2000 (hereinafter Bachmann et al.); and

whether claims 14, 15, 19, and 20 are unpatentable over Bachmann et al. in view of Kennedy.

7. GROUPING OF CLAIMS

The claims stand and fall together as follows:

Group A -- claims 1-3, 6-8, and 16;

Group B -- claims 4, 5, 9-13, and 17-20;

Group C -- claims 14 and 15.

8. ARGUMENTS

Argument 8.A.

Was 35 U.S.C. § 103(a) properly applied in a rejection of claims 1-3 and 6-8 (Group A) as being unpatentable over Kennedy in view of Bachmann et al.?

Arguments in support of patentability

With respect to the grouping of claims for arguments in support of patentability, dependent claims 2, 3, and 6-8 depend from claim 1. Independent claim 1 is a method claim,

and independent claim 16 is directed to a computer program product. Although these independent claims are similar, claim 1 is broader. Hence, for purposes of this argument, Appellant argues for the patentability of the present invention using  
5 claim 1 as an exemplary claim.

Rejections under 35 U.S.C. 103 must provide a *prima facie* case for obviousness. According to 37 C.F.R. § 1.192(c)(8)(iv), for each rejection under 35 U.S.C. § 103, Appellant must specify the errors in the rejection, the  
10 specific limitations in the rejected claims which are not described in the prior art relied on in the rejection, and how such limitations render the claimed subject matter nonobvious over the prior art. If the rejection is based upon a combination of references, the argument shall explain why the  
15 references, taken as a whole, do not suggest the claimed subject matter. In summary, Appellant argues that the pending claims in the present patent application are patentable because the rejection of the independent claim 1 fails to provide a *prima facie* case of obviousness. Independent claim 1 reads:

20 1. A method for deleting entries from a directory in which directory information is stored in a set of database tables, comprising the steps of:  
receiving a request to delete a directory entry;  
responsive to receiving the request to delete a  
25 directory entry, determining to tag the directory entry for subsequent deletion by setting an attribute of the directory entry to a predetermined value;  
updating a first database table storing the attribute of the directory entry;  
30 periodically searching for tagged directory entries in the first database table during a cleanup process interval; and  
deleting references to the tagged directory entries throughout the set of database tables.

Rejection of claim 1

The rejection of independent claim 1 is structured as follows. The rejection asserts that a portion of the second  
5 element of claim 1, i.e. "determining to tag the directory entry for subsequent deletion by setting an attribute of the directory entry to a predetermined value", is shown in Kennedy at column 9, lines 50-62, which reads:

10 The database 39 can include multiple data fields, organized within an array structure, for maintaining message related information. To support download and delete operations, typical data fields of the database include: a session identifier (session ID) 200, a unique  
15 identifier (UIDL) 205, a message size 210, an entry identifier (EID) time, an "on server" flag 225, a download flag 230, and a ["delete" flag 235. Message re-assembly operations can be] supported by adding to the database structure certain data fields corresponding to portions of a MIME-compatible message, such as a message  
20 group identifier (message group ID) 240, a message part number 245, and a total part number 250.

The rejection asserts that the fourth element of claim 1, i.e. "periodically searching for tagged directory entries in the  
25 first database table during a cleanup process interval", is shown in Kennedy at column 11, lines 55-64, which reads:

30 The user also can set a time-based parameter in the message manager program module 37 that define a time period for expiring all messages maintained in the local message store 38 after pre-defined time period. Because the database 39 maintains local time entries for each message, those messages satisfying the pre-defined time period for expiration can be marked with a "delete" flag. In this example, the user has set a seventy-two hour time  
35 period for expiration of a message and its associated message entry.

The rejection asserts that the fifth element of claim 1, i.e. "deleting references to the tagged directory entries throughout the set of database tables", is shown in Kennedy at column 11, line 65, to column 12, line 3, which reads:

5           Message entries containing "delete" flags in the  
          database 39 can be deleted from the server 49 after all  
          message retrieval operations are completed. In  
          particular, all message entries marked with a "delete"  
10          flag are located in response to walking the message  
          entries in the database 39 and thereafter, are deleted  
          from the server 49.

The rejection then admits that "Kennedy does not disclose receiving a request to delete a directory entry; responsive to  
15 receiving the request to delete a directory entry; and  
updating a first database table storing the attribute of the directory value". In other words, Kennedy does not disclose the first element, part of the second element, and the third element of claim 1. The rejection then turns to Bachmann et  
20 al. as disclosing the elements of claim 1 that are not disclosed by Kennedy.

The rejection asserts that the first element of claim 1 and part of the second element, i.e. "receiving a request to delete a directory entry" and "responsive to receiving the  
25 request to delete a directory entry", is shown in Bachmann et  
al. at column 3, lines 48-61, which reads:

30          A client machine 10 makes a TCP/IP connection to an LDAP server 12 through network 11, sends requests and receives responses. LDAP server 12 supports a directory 21 as illustrated in a simplified form in FIG. 2. Each of the client and server machines further include a directory "runtime" environment 25 for implementing the directory service operations as will be described below. The directory 21 is based on the concept of an "entry" 27,  
35          which contains information about some object (e.g., a

person). Entries are composed of attributes 29, which have a type and one or more values. Each attribute 29 has a particular syntax that determines what kinds of values are allowed in the attributes (e.g., ASCII text, binary characters, and the like) and how these values are constrained during a particular directory operation.

The rejection asserts that the third element of claim 1, i.e. "updating a first database table storing the attribute of the directory value", is shown in Bachmann et al. at column 6, lines 60-68, which reads:

FIG. 8 is a flowchart for a routine called ldap\_add for adding entries to the database. Because the directory structure will be changed when entries are added into the database, the parent table (or ldap\_entry) and the descendant table (ldap\_desc) are updated to reflect the change. In other words, after all tables get created, the ldap\_add routine is used to populate the tables with correct information.

The rejection then concludes with the following motivation statement:

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Kennedy with the teachings of Bachmann. By doing so, the system provides a faster search and more efficient, (col. 2, lines 9-10).

#### Analysis of the rejection

The rejection of the claim 1 under Kennedy and Bachmann et al. starts with an attempt to build a foundation for an obviousness argument with multiple citations to passages of Kennedy; these passages are applied against a portion of claim 1. However, none of these passages disclose the claimed features of the independent claims, as explained in more detail hereinbelow.



As explained in the "Summary of the Invention" section of the Kennedy reference, the system that is disclosed in Kennedy provides a client-server method for managing electronic mail messages that are stored within a local, client-side database and a remote, server-side database. During a client-server session, message-related information is retrieved from the server and stored in the client-based database along with downloaded messages. Indications are provided in the client-side database as to whether a message has been previously downloaded from the server and whether a copy of a message has been left on the server. Expiration times can be set so that messages are deleted over time. In essence, the system allows some of the information to be stored at the client and/or the server, and the system acts to synchronize the information in the local database and the remote database.

In contrast, the present invention is a method for deleting an entry from a directory in which directory information is stored in a set of database tables; the deletion is initiated in response to a request to delete a directory entry. In response, the directory entry is tagged in some manner as being a deleted entry. At a periodic interval, the routine then searches for tagged entries, and references to the tagged entries are then deleted throughout the set of database tables.

More specifically, the rejection is misleading in its treatment of the actual claim language. It is important to note that all of the claims in the present application are directed to a directory service or a method or a computer program product that involves a directory, but Kennedy does

not even mention a directory or a directory service. Since Kennedy does not disclose a directory or a directory service, it is not possible for Kennedy to disclose the claim elements as asserted by the rejection. For example, it is not possible  
5 for Kennedy to disclose a portion of the second element of claim 1, i.e. "determining to tag the directory entry for subsequent deletion by setting an attribute of the directory entry to a predetermined value", when Kennedy does not disclose any type of directory. The rejection does not  
10 explain how or why Kennedy discloses the claim element; Appellant asserts that the rejection lacks any argument on this point because the lack of disclosure in Kennedy makes the argument indefensible.

In addition, the rejection appears to draw an analogy  
15 between tagging message entries with delete flags in Kennedy and the tagging operation in the present invention. However, the rejection relies on Bachmann et al. to disclose an operation "responsive to a request to delete a directory entry" as required by the claim language. Thus, the rejection  
20 bifurcates the second claim element; the reason for performing the method step in the second element of claim 1 is supposedly shown in Bachmann et al. while the subsequent action is supposedly shown in Kennedy. The rejection does not make any attempt to explain how a request to delete a directory entry  
25 could be incorporated into the system disclosed in Kennedy or why this request would be useful.

Appellant notes that the client-server system in Kennedy operates in a particular way to automatically synchronize messages in the client database and the server database. To

somehow allow a user at the client to request deletion of a message on the server would completely change the principle of operation of the system in Kennedy. In allowing such an operation, the client database and the server database would become unsynchronized, and the server would be unable to subsequently determine which messages on the client refer to which messages on the server, and vice versa. As noted in MPEP § 2143.01:

If the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims prima facie obvious. In re Ratti, 270 F.2d 810, 12 USPQ 349 (CCPA 1959).

As shown by Appellant in the arguments above, Kennedy and Bachmann et al. do not disclose the claim elements as stated in the rejection. The rejection then proceeds to state that it would have been obvious to combine features from the system disclosed in Kennedy and the system disclosed in Bachmann et al.. However, the motivational statement in the rejection is inaccurate and completely generic: "By doing so, the system provides a faster search and more efficient."

First, the motivational statement is inaccurate. Assuming *arguendo* that Kennedy and Bachmann et al. could be combined to reach the present invention, any search through a database that contains entries that have been processed using the delayed deletion mechanism of the present invention would necessarily provide a slower search. The steps to check for tagged entries requires extra processing during the search, thereby necessarily producing a slower search.

Second, the motivational statement is completely generic.

The rejection does not provide any argument as to how or why a characteristic of the system of Kennedy or a characteristic of the system of Bachmann et al. would motivate one of  
5 ordinary skill in the art to combine any feature of these systems with any feature of the other system. Appellant admits that, in general, one of ordinary skill in the art would desire a faster and more efficient system. However, Appellant asserts that the rejection completely fails to  
10 provide a reason why one would be motivated to combine specific features from the prior art references.

Bachmann et al. does not disclose the features that Kennedy fails to disclose, as discussed above. Therefore, both Kennedy and Bachmann et al. both fail to disclose the  
15 claimed elements of claim 1. In addition, there is no motivation or suggestion in the prior art for combining features from these references, and even if one were motivated to do so, the system that is disclosed in Bachmann et al. could not be integrated into the system that is disclosed in  
20 Kennedy to form the claimed features that are not shown in Kennedy as asserted by the rejection.

Examiner bears the burden of establishing a prima facie case of obviousness

25 Contrary to the assertions of the rejection, Kennedy clearly fails to show claimed features of the present invention. Moreover, both Kennedy and the combination of Bachmann et al. and Kennedy fail to show the claimed features. As should be recognized, because both the primary and  
30 secondary references in the rejection fail to disclose the

claimed features against which the references were applied, and because the references fail to be combinable to produce these features, the rejection fails to fulfill the requirements of a proper obviousness argument.

5       Hence, a rejection of claim 1 cannot be based upon the cited prior art to establish a *prima facie* case of obviousness.

          Therefore, a rejection of the claim 1 under 35 U.S.C. § 103(a) has been shown to be insupportable in view of the cited prior art, and claim 1 is patentable over the applied

10       references. For this and other reasons, Appellant argues that the position of the Examiner should be reversed and that the rejection of claim 1 should not be upheld.

Argument 8.B.

15       Was 35 U.S.C. § 103(a) properly applied in a rejection of claims 4, 5, 9-13, and 17-20 (Group B) as being unpatentable over Kennedy in view of Bachmann et al.?

Arguments in support of patentability

20       With respect to the grouping of claims for arguments in support of patentability, Appellant argues for the patentability of the present invention using independent claim 9 as an exemplary claim. Independent claim 9 is similar to independent claim 1, which was discussed with respect to claim

25       Group A above, except that claim 9 includes an additional method step; claims 10-13 depend from claim 9. A similar additional method step is recited in dependent claim 4 (which depends from claim 1 in Group A that was discussed above), and a similar additional means-for element is recited in dependent

claim 17 (which depends from claim 16 in Group A); claim 5 depends from claim 4, and claim 18 depends from claim 17. Independent claim 19 is similar to independent claim 9 except that claim 19 is directed to a directory service whereas claim 9 is directed to a method; claim 20 depends from claim 19. Hence, for purposes of this argument, claim 9 is an appropriate exemplary claim.

In summary, Appellant argues that the pending claims in the present patent application are patentable because the rejection of the independent claim 9 fails to provide a *prima facie* case of obviousness. The fourth element of independent claim 9 is not included in independent claim 1; independent claim 9 reads:

9. A method for deleting entries from a directory in which directory information is stored in a set of database tables, comprising the steps of:  
receiving a request to delete a directory entry;  
responsive to receiving the request to delete a directory entry, determining to tag the directory entry for subsequent deletion by setting an attribute of the directory entry to a predetermined value;  
updating a first database table storing the attribute of the directory entry;  
responsive to a search for directory entries that satisfy a search query, excluding tagged directory entries from search results that otherwise satisfy the search query;  
periodically searching for tagged during entries during a cleanup process interval; and  
deleting references to the tagged directory entries throughout the set of database tables.

#### Rejection of claim 9

The rejection of independent claim 9 relies on the argument that was provided in the rejection of independent claim 1 and then extends that argument with additional

material to address the additional method step in claim 9 that is not present in claim 1. Thus, the argument that was provided by Appellant for claim 1 above is equally applicable to claim 9. Appellant hereinbelow extends Appellant's  
5 arguments with respect to the additional method step in claim 9 that is not present in claim 1.

The rejection of claim 9 asserts that the additional element of claim 9, i.e. "responsive to a search for directory entries that satisfy a search query, excluding tagged  
10 directory entries from search results that otherwise satisfy the search query", is shown in Bachmann et al. at claim 1 and at column 5, line 60, to column 6, line 1.

Claim 1 of Bachmann et al. reads as follows:

1. A method of searching a directory organized as a  
15 naming hierarchy having a plurality of entries each represented by a unique identifier, comprising the steps of:

generating a relational table associating unique  
20 identifier pairs in the naming hierarchy having a given hierarchical relationship;

in response to a search query having a given filter  
criteria and search scope, returning a list of entries  
that satisfy the given filter criteria; and

25 using the relational table to filter out entries in  
the list according to the given search criteria.

The portion of Bachmann et al. at column 5 reads:

LDAP provides a number of known functions including  
30 query (search and compare), update, authentication and others. The search and compare operations are used to retrieve information from the database. For the search function, the criteria of the search is specified in a search filter. The search filter typically is a Boolean expression that consists of attribute name, attribute  
35 value, and Boolean operations like AND, OR and NOT. Users can use the filter to perform complex search operations.

Analysis of the rejection

The rejection of the claim 9 under Kennedy and Bachmann et al. apparently relies on the disclosure of the search filters in Bachmann et al. as disclosing the fourth element of claim 9, i.e. "responsive to a search for directory entries that satisfy a search query, excluding tagged directory entries from search results that otherwise satisfy the search query". The search filters in Bachmann et al. cannot be equated with the step of excluding tagged directory entries. During the execution of the search function, the search filter would be applied against the information in the directory entries; after comparing the values in the search filter with the information in the directory entry, the search filter function determines whether or not the directory entry satisfies the search filter function, includes or does not include the directory entry in the results, and then proceeds to the next directory entry. In examining the information in the directory entry, there would be no other processing steps of looking to determine whether the directory entry information somehow indicates that the directory entry is tagged as an entry that is waiting for subsequent deletion, as is required by the claim language of the present invention.

In other words, while Bachmann et al. clearly discloses the first portion of the fourth element, i.e. "responsive to a search for directory entries that satisfy a search query", Bachmann et al. clearly cannot disclose the second portion of the fourth element, i.e. "excluding tagged directory entries from search results that otherwise satisfy the search query". Since Bachmann et al. does not disclose the step of tagging



entries for subsequent deletion, which is recited as the second element of claim 9, Bachmann et al. cannot disclose a step of excluding those tagged directory entries.

Furthermore, the rejection does not state why it would  
5 have been obvious to combine features from the system disclosed in Kennedy and the system disclosed in Bachmann et al.. There is no motivational statement in the rejection. Clearly, the rejection fails to provide a *prima facie* case of obviousness when there is no argument as to why one having  
10 ordinary skill in the art would have been motivated in an obvious manner to combine features in the prior art references.

Bachmann et al. does not disclose the features that Kennedy fails to disclose, as discussed above. Therefore,  
15 both Kennedy and Bachmann et al. both fail to disclose the claimed elements of claim 9. In addition, there is no motivation or suggestion in the prior art for combining features from these references, and even if one were motivated to do so, the system that is disclosed in Bachmann et al.  
20 could not be integrated into the system that is disclosed in Kennedy to form the claimed features that are not shown in Kennedy, as was discussed by Appellant with respect to the rejection of independent claim 1.

25 Examiner bears the burden of establishing a *prima facie* case of obviousness

Contrary to the assertions of the rejection, Kennedy clearly fails to show claimed features of the present invention. Moreover, both Kennedy and the combination of

Bachmann et al. and Kennedy fail to show the claimed features. As should be recognized, because both the primary and secondary references in the rejection fail to disclose the claimed features against which the references were applied, and because the references fail to be combinable to produce these features, the rejection fails to fulfill the requirements of a proper obviousness argument.

Hence, a rejection of claim 9 cannot be based upon the cited prior art to establish a *prima facie* case of obviousness.

Therefore, a rejection of the claim 9 under 35 U.S.C. § 103(a) has been shown to be insupportable in view of the cited prior art, and claim 9 is patentable over the applied references. For this and other reasons, Appellant argues that the position of the Examiner should be reversed and that the rejection of claim 9 should not be upheld.

#### Argument 8.C.

Was 35 U.S.C. § 103(a) properly applied in a rejection of claims 14 and 15 (Group C) as being unpatentable over Bachmann et al. in view of Kennedy?

#### Arguments in support of patentability

With respect to the grouping of claims for arguments in support of patentability, Appellant argues for the patentability of the present invention using independent claim 14 as an exemplary claim; claim 15 depends from claim 14. Independent claim 14 differs from the other independent claims in that claim 14 is directed solely to a method for searching a database, whereas the other independent claims are directed

to a method (and corresponding computer program product, etc.)  
for deleting directory entries in a specific manner, e.g.,  
claim 1, or for deleting directory entries and searching  
directory entries in a specific manner, e.g., claim 9. Hence,  
5 for purposes of this argument, claim 14 is an appropriate  
exemplary claim. In summary, Appellant argues that the pending  
claims in the present patent application are patentable because  
the rejection of the independent claim 14 fails to provide a  
*prima facie* case of obviousness.

10 Independent claim 14 reads:

14. A method for searching a database from a directory  
service, comprising the steps of:

receiving a search query;

15 responsive to a search for directory entries that  
satisfy the search query, excluding given directory  
entries from search results that otherwise satisfy the  
search query, wherein a given directory entry is a  
directory entry that has been tagged for deletion by  
setting an attribute of the given directory entry to a  
20 predetermined value; and

returning the search results.

#### Rejection of claim 14

The rejection of independent claim 14 is similar to a  
25 combination of the rejection of claim 9 and a portion of the  
rejection of claim 1. The rejection of claim 14 asserts that  
a portion of the second element of claim 14, i.e. "responsive  
to a search for directory entries that satisfy a search query,  
excluding tagged directory entries from search results that  
30 otherwise satisfy the search query", is shown in Bachmann et  
al. at claim 1 and at column 5, line 60, to column 6, line 1,  
as was also argued with respect to claim 9 because a portion

of the second element of claim 14 is similar to the fourth element of claim 9.

Analysis of the rejection

5 Appellant does not argue against the fact that Bachmann et al. discloses the first and third elements of claim 14. Bachmann et al. discloses "receiving a search query" and "returning the search results".

10 However, as argued by Appellant above with respect to claim 9, the rejection of the claim 14 under Kennedy and Bachmann et al. apparently relies on the disclosure of the search filters in Bachmann et al. as disclosing the portion of the second element of claim 14, i.e. "responsive to a search for directory entries that satisfy a search query, excluding  
15 tagged directory entries from search results that otherwise satisfy the search query". The search filters in Bachmann et al. cannot be equated with the step of excluding tagged directory entries. During the execution of the search function, the search filter would be applied against the  
20 information in the directory entries; after comparing the values in the search filter with the information in the directory entry, the search filter function determines whether or not the directory entry satisfies the search filter function, includes or does not include the directory entry in  
25 the results, and then proceeds to the next directory entry. In examining the information in the directory entry, there would be no other processing steps of looking to determine whether the directory entry information somehow indicates that the directory entry is tagged as an entry that is waiting for

subsequent deletion, as is required by the claim language of the present invention.

In other words, while Bachmann et al. clearly discloses the first portion of the second element of claim 14, i.e.

5 "responsive to a search for directory entries that satisfy a search query", Bachmann et al. clearly cannot disclose the second portion of the second element, i.e. "excluding tagged directory entries from search results that otherwise satisfy the search query". Since Bachmann et al. does not disclose  
10 the step of tagging entries for subsequent deletion, which is recited as the third portion of the second element of claim 14, Bachmann et al. cannot disclose a step of excluding those tagged directory entries.

Furthermore, Appellant's argument is supported by the  
15 argument in the rejection. The rejection admits that Bachmann et al. does not disclose the step of tagging entries for subsequent deletion. The rejection states: "Bachmann does not disclose wherein a given directory entry is a directory entry that has been tagged for deletion by setting an attribute of  
20 the given directory entry to a predetermined value." Again, if Bachmann et al. does not disclose the step of tagging entries for subsequent deletion, Bachmann et al. cannot disclose a step of excluding those tagged directory entries.

At this point, the rejection provides the same argument  
25 as was given for a similar element in claim 1 by relying on Kennedy as disclosing the feature of "wherein a given directory entry is a directory entry that has been tagged for deletion by setting an attribute of the given directory entry to a predetermined value". As Appellant argued above with

respect to claim 1, all of the claims in the present application are directed to a directory service or a method or a computer program product that involves a directory, but Kennedy does not even mention a directory or a directory service. Since Kennedy does not disclose a directory or a directory service, it is not possible for Kennedy to disclose the claim elements as asserted by the rejection. For example, it is not possible for Kennedy to disclose a portion of the second element of claim 14, i.e. "wherein a given directory entry is a directory entry that has been tagged for deletion by setting an attribute of the given directory entry to a predetermined value", when Kennedy does not disclose any type of directory. The rejection does not explain how or why Kennedy discloses the claim element; Appellant asserts that the rejection lacks any argument on this point because the lack of disclosure in Kennedy makes the argument indefensible.

As shown by Appellant in the arguments above, Bachmann et al. and Kennedy do not disclose the claim elements as stated in the rejection. The rejection then proceeds to state that it would have been obvious to combine features from the system disclosed in Bachmann et al. and the system disclosed in Kennedy. However, the motivational statement in the rejection is irrelevant and completely generic: "By doing so, the business increasingly rely [sic] on e-mail messages to share ideas, transmit documents, schedule meeting, and perform a multitude of other everyday tasks."

First, the motivational statement is irrelevant. Assuming *arguendo* that Bachmann et al. and Kennedy could be combined to reach the present invention, it is entirely

unclear why one having ordinary skill in the art would have been motivated to do so merely because businesses increasingly rely on the use of e-mail messages.

Second, the motivational statement is completely generic.

5 The rejection does not provide any argument as to how or why a characteristic of the system of Bachmann et al. or a characteristic of the system of Kennedy would motivate one of ordinary skill in the art to combine any feature of these systems with any feature of the other system. Appellant  
10 admits that, in general, one of ordinary skill in the art would desire a better e-mail system. However, Appellant asserts that the rejection completely fails to provide a reason why one would be motivated to combine specific features from the prior art references.

15 Kennedy does not disclose the features that Bachmann et al. fails to disclose, as discussed above. Therefore, both Bachmann et al. and Kennedy both fail to disclose the claimed elements of claim 14. In addition, there is no motivation or suggestion in the prior art for combining features from these  
20 references, and even if one were motivated to do so, the system that is disclosed in Kennedy could not be integrated into the system that is disclosed in Bachmann et al. to form the claimed features that are not shown in Bachmann et al., as was discussed by Appellant with respect to the rejection of  
25 independent claim 1.

Examiner bears the burden of establishing a prima facie case of obviousness

Contrary to the assertions of the rejection, Bachmann et al.  
30 al. clearly fails to show claimed features of the present

invention. Moreover, both Bachmann et al. and the combination of Bachmann et al. and Kennedy fail to show the claimed features. As should be recognized, because both the primary and secondary references in the rejection fail to disclose the claimed features against which the references were applied, and because the references fail to be combinable to produce these features, the rejection fails to fulfill the requirements of a proper obviousness argument.

Hence, a rejection of claim 14 cannot be based upon the cited prior art to establish a *prima facie* case of obviousness.

Therefore, a rejection of the claim 14 under 35 U.S.C. § 103(a) has been shown to be insupportable in view of the cited prior art, and claim 14 is patentable over the applied references. For this and other reasons, Appellant argues that the position of the Examiner should be reversed and that the rejection of claim 14 should not be upheld.



9. Conclusion

In view of the above arguments, it is respectfully urged that the rejections of the claims should not be sustained.

5 DATE: December 4, 2003

Respectfully submitted,



Joseph R. Burwell

Reg. No. 44,468

10

ATTORNEY FOR APPELLANT

Law Office of Joseph R. Burwell

P.O. Box 28022

Austin, Texas 78755-8022

15

Voice: 866-728-3688 (866-PATENT8)

Fax: 866-728-3680 (866-PATENT0)

Email: joe@burwell.biz

## 10. APPENDIX OF CLAIMS

1. A method for deleting entries from a directory in which  
5 directory information is stored in a set of database tables,  
comprising the steps of:

receiving a request to delete a directory entry;

responsive to receiving the request to delete a directory  
entry, determining to tag the directory entry for subsequent  
10 deletion by setting an attribute of the directory entry to a  
predetermined value;

updating a first database table storing the attribute of  
the directory entry;

periodically searching for tagged directory entries in  
15 the first database table during a cleanup process interval;  
and

deleting references to the tagged directory entries  
throughout the set of database tables.

20 2. The method as described in claim 1 wherein the directory  
entry is tagged by setting its creation time attribute to a  
given value.

3. The method as described in claim 2 wherein the given  
25 value is a null value.

4. The method as described in claim 1, further including the steps of:

performing a search for directory entries that satisfy a search query; and

5 excluding tagged directory entries from search results that otherwise satisfy the search query.

5. The method as described in claim 4 wherein the step of excluding tagged directory entries includes modifying an SQL  
10 query to exclude rows having a null change creation.

6. The method as described in claim 1 wherein the directory is a Lightweight Directory Access Protocol (LDAP) directory service and the database tables are managed by a relational  
15 database management service.

7. The method as described in claim 1 wherein the first database table is an entry table.

20 8. The method as described in claim 7 wherein the set of database tables includes at least one attribute table storing information about an attribute.

9. A method for deleting entries from a directory in which directory information is stored in a set of database tables, comprising the steps of:

receiving a request to delete a directory entry;

5 responsive to receiving the request to delete a directory entry, determining to tag the directory entry for subsequent deletion by setting an attribute of the directory entry to a predetermined value;

10 updating a first database table storing the attribute of the directory entry;

responsive to a search for directory entries that satisfy a search query, excluding tagged directory entries from search results that otherwise satisfy the search query;

15 periodically searching for tagged entries during a cleanup process interval; and

deleting references to the tagged directory entries throughout the set of database tables.

10. The method as described in claim 9 wherein the directory entry is tagged by setting its creation time to a given value.

11. The method as described in claim 10 wherein the given value is a null value.

25 12. The method as described in claim 9 wherein the first database table is an entry table.

13. The method as described in claim 12 wherein the set of database tables includes at least one attribute table storing information about an attribute.

5 14. A method for searching a database from a directory service, comprising the steps of:  
receiving a search query;  
responsive to a search for directory entries that satisfy the search query, excluding given directory entries from  
10 search results that otherwise satisfy the search query,  
wherein a given directory entry is a directory entry that has been tagged for deletion by setting an attribute of the given directory entry to a predetermined value; and  
returning the search results.

15 15. The method as described in claim 14 where in the directory service is a Lightweight Directory Access Protocol (LDAP) directory service and the database tables are managed by a relational database management service.

20

16. A computer program product in a computer-readable medium for deleting entries from a directory in which directory information is stored in a set of database tables, comprising:

- means for receiving a request to delete a directory entry;
- means for determining, responsive to receiving the request to delete a directory entry, to tag the directory entry for subsequent deletion by setting an attribute of the directory entry to a predetermined value;
- means for updating a first database table storing the attribute of the directory entry;
- means for periodically searching for tagged directory entries in the first database table during a cleanup process interval; and
- means for deleting references to the tagged directory entries throughout the set of database tables.

17. The computer program product as described in claim 16, further including:

- means responsive to a search for directory entries that satisfy a search query for excluding tagged entries from search results that otherwise satisfy the search query.

18. The computer program product as described in claim 17 wherein the search query is a Lightweight Directory Access Protocol (LDAP) directory service query.

19. A directory service, comprising:

a directory organized as a naming hierarchy having a plurality of entries each represented by a unique identifier;

a relational database management system having a backing  
5 store for storing directory data in a set of database entries;  
and

means for deleting entries from the directory,  
comprising:

means for determining, responsive to receiving the  
10 request to delete a directory entry, to tag the directory  
entry for subsequent deletion by setting an attribute of  
the directory entry to a predetermined value;

means for updating a first database table storing  
the attribute of the directory entry;

15 means for periodically searching for tagged  
directory entries in the first database table during a  
cleanup process interval; and

means for deleting references to the tagged  
directory entries throughout the set of database tables;  
20 and

means responsive to a search for directory entries  
that satisfy a search query for excluding tagged  
directory entries from search results that otherwise  
satisfy the search query.

25

20. The directory service as described in claim 19 wherein  
the directory is compliant with the Lightweight Directory  
Access Protocol (LDAP).